

## IMAGE PROCESSING SYSTEM

Publication number: JP10255069

Publication date: 1998-09-25

Inventor: KUMAGAI ATSUSHI; UCHIYAMA SHINJI; KATAYAMA AKIHIRO

Applicant: CANON KK

Classification:

- international: G06T17/40; G06T15/00; G06T17/40; G06T15/00;  
(IPC1-7): G06T15/00

- European: G06T15/00

Application number: JP19970054782 19970310

Priority number(s): JP19970054782 19970310

Also published as:

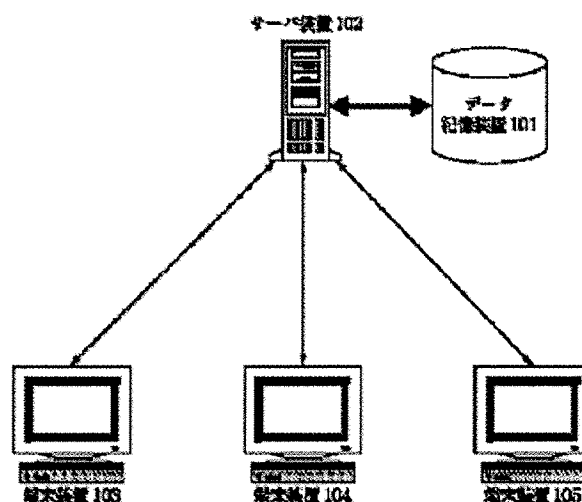
EP0865001 (A2)  
US6151028 (A1)  
EP0865001 (A3)  
EP0865001 (B1)

Report a data error here

### Abstract of JP10255069

**PROBLEM TO BE SOLVED:** To share a virtual environment that includes not only simple shape data but also an object which has a very complicated shape among plural users.

**SOLUTION:** A terminal 103 displays a virtual environment from the point of view of a user of the terminal 103 based on the data of a tree structure of which constituent elements are shape data, light space data, etc., and sends the data of the tree structure to a server 102, the server 102 stores received data in a data storage device 101 and then sends it to a terminal 104, and the terminal 104 displays a virtual environment from the point of view of a user of the terminal 104, based on the received data of the tree structure.



Data supplied from the **esp@cenet** database - Worldwide

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-255069

(43) 公開日 平成10年(1998) 9月25日

(51) Int.Cl.<sup>6</sup>  
G 0 6 T 15/00

識別記号

F I  
G 0 6 F 15/62

3 6 0

審査請求 未請求 請求項の数12 O L (全 29 頁)

(21) 出願番号 特願平9-54782

(22) 出願日 平成9年(1997) 3月10日

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 熊谷 篤

東京都大田区下丸子3丁目30番2号キヤノ  
ン株式会社内

(72) 発明者 内山 晋二

東京都大田区下丸子3丁目30番2号キヤノ  
ン株式会社内

(72) 発明者 片山 昭宏

東京都大田区下丸子3丁目30番2号キヤノ  
ン株式会社内

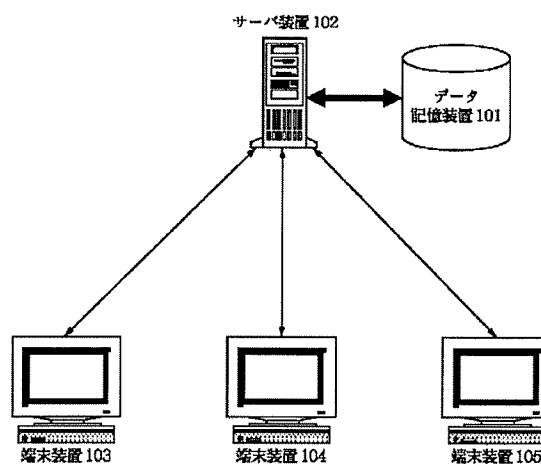
(74) 代理人 弁理士 丸島 儀一

(54) 【発明の名称】 画像処理システム

(57) 【要約】

【課題】 単純な形状データだけではなく、形状が非常に複雑な物体を含む仮想環境を、複数の利用者が共有することができるようにする。

【解決手段】 端末装置103は、形状データと光線空間データ等を構成要素とする木構造のデータに基づいて、該端末装置103のユーザーから見た場合の仮想環境を表示して、木構造のデータをサーバ102に送信し、サーバ102は、受信したデータをデータ記憶装置101に格納してから端末装置104に送信し、端末装置104は、受信した木構造のデータに基づいて、端末装置104のユーザーから見た場合の仮想環境を表示する。



【特許請求の範囲】

【請求項1】 サーバと、該サーバに接続された複数の端末とからなる画像処理システムであって、第1の端末は、3次元的位置と3次元の向きと表面属性とを含む記述情報に基づいて、3次元物体の形状や3次元空間の形状を記述する第1の記述手段と、任意の視点からの3次元物体と3次元空間の見え方によって、実写画像に基づいて、該3次元物体と該3次元空間を記述する第2の記述手段と、前記第1の記述手段と前記第2の記述手段による記述内容を構成要素とする木構造のデータを記憶する第1の記憶手段と、前記第1の記憶手段に記憶された木構造のデータに基づいて、仮想3次元空間の第1の位置から第1の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示する第1の表示手段と、前記第1の記憶手段に記憶された木構造のデータを、前記サーバに送信する第1の送信手段とを有し、前記サーバは、前記第1の送信手段により送信されたデータを受信する第1の受信手段と、前記第1の受信手段により受信されたデータを記憶する第2の記憶手段と、前記記憶手段に記憶されたデータを、前記サーバに接続されている端末に送信する第2の送信手段とを有し、第2の端末は、前記第2の送信手段により送信されたデータを受信する第2の受信手段と、前記第2の受信手段により受信されたデータを記憶する第3の記憶手段と、前記第3の記憶手段に記憶された木構造のデータに基づいて、仮想3次元空間の第2の位置から第2の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示する第2の表示手段とを有することを特徴とする画像処理システム。

【請求項2】 前記第1の端末は、前記第1の記憶手段に記憶された木構造のデータの構成要素の少なくとも一部に処理を施す処理手段と、前記処理手段による処理の内容に基づいて、前記第1の記憶手段に記憶された木構造のデータを変更する変更手段とを更に有し、前記第1の端末において、前記第1の送信手段は、前記変更手段により変更された木構造のデータを、前記サーバに送信し、前記第2の端末において、前記第2の表示手段は、前記変更手段により変更された木構造のデータに基づいて、仮想3次元空間の第2の位置から第2の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示することを特

徴とする請求項1記載の画像処理システム。

【請求項3】 前記第1の端末において、前記第1の送信手段は、前記第1の記憶手段に記憶された木構造のデータとともに、仮想3次元空間の第1の位置から第1の方向を見ている利用者を仮想して、これを前記第1の端末を利用している第1の利用者として、該第1の利用者の情報を前記サーバに送信することを特徴とする請求項1記載の画像処理システム。

【請求項4】 前記第2の端末において、前記第2の受信手段は、前記第2の送信手段により送信された木構造のデータとともに、前記第1の利用者の情報を、前記サーバから受信し、前記第2の表示手段は、前記第1の記述手段と前記第2の記述手段による記述内容とともに、前記第1の利用者を示すシンボルを表示することを特徴とする請求項3記載の画像処理システム。

【請求項5】 前記第1の端末において、前記第2の記述手段は、光線空間データを用いて、前記3次元物体と前記3次元空間を記述することを特徴とする請求項1記載の画像処理システム。

【請求項6】 前記第1の端末において、前記処理手段は、前記第1の記憶手段に記憶された木構造のデータの構成要素の少なくとも一部を移動または追加または削除または変更することを特徴とする請求項2記載の画像処理システム。

【請求項7】 前記第1の端末において、前記第1の表示手段は、前記処理手段によって前記第1の記憶手段に記憶された木構造のデータの構成要素の少なくとも一部に処理を施す際に、前記処理手段の処理対象となる構成要素を識別可能に表示し、前記第1の送信手段は、前記第1の記憶手段に記憶された木構造のデータとともに、前記処理手段の処理対象となる構成要素の情報を前記サーバに送信することを特徴とする請求項2記載の画像処理システム。

【請求項8】 前記第2の端末において、前記第2の受信手段は、前記第2の送信手段により送信された木構造のデータとともに、前記処理手段の処理対象となる構成要素の情報を、前記サーバから受信し、前記第2の表示手段は、前記第1の記述手段と前記第2の記述手段による記述内容とともに、前記処理手段の処理対象となる構成要素を識別可能に表示することを特徴とする請求項7記載の画像処理システム。

【請求項9】 請求項1から8に記載の第1の端末。

【請求項10】 請求項1から8に記載のサーバ。

【請求項11】 請求項1から8に記載の第2の端末。

【請求項12】 複数の端末が相互に接続された画像処理システムであって、第1の端末は、

3次元的位置と3次元の向きと表面属性とを含む記

述情報に基づいて、3次元物体の形状や3次元空間の形状を記述する第1の記述手段と、  
任意の視点からの3次元物体と3次元空間の見え方に従って、実写画像に基づいて、該3次元物体と該3次元空間を記述する第2の記述手段と、  
前記第1の記述手段と前記第2の記述手段による記述内容を構成要素とする木構造のデータを記憶する第1の記憶手段と、  
前記第1の記憶手段に記憶された木構造のデータに基づいて、仮想3次元空間の第1の位置から第1の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示する第1の表示手段と、  
前記第1の記憶手段に記憶された木構造のデータを、他の端末に送信する送信手段とを有し、  
第2の端末は、  
前記送信手段により送信されたデータを受信する受信手段と、  
前記受信手段により受信されたデータを記憶する第2の記憶手段と、  
前記第2の記憶手段に記憶された木構造のデータに基づいて、仮想3次元空間の第2の位置から第2の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示する第2の表示手段とを有することを特徴とする画像処理システム。

#### 【発明の詳細な説明】

#### 【0001】

【発明の属する技術分野】本発明は、バーチャルリアリティのための仮想環境を生成し、複数の利用者間で、生成された仮想環境を共有する画像処理システムに関するものである。

#### 【0002】

【従来の技術】従来、3次元物体や空間を表現し、任意の位置・向きからの観察画像を提示する手法として、  
(1)ポリゴンデータや曲面データなどの形状モデルデータ、その表面属性や模様を表すテクスチャデータ、光源データなどによって、3次元物体や空間を表現し、コンピュータグラフィックスのレンダリング技術により任意の位置向きでの空間の見え方を画面に描画することによって、3次元空間や物体を再現する方法があった。

【0003】この方法により3次元の仮想環境を構築する際に、仮想環境を構成する要素（座標変換データ、形状データ、表面属性データ、照明など）を木構造により表現する方法があった。これは、3次元空間を構成する空間、地面、建物、部屋、家具、照明、置物などは、それぞれもともと階層的な入れ子関係にあると考えられるためである。例えば、机の上の置物は、机の配置とともに動くように、机の配置に依存する関係にあり、机の存在する座標系から相対的に配置したほうが都合がよい場合が多い。そのため、階層的に配置に依存関係のあるデータ構造をとる。その方法として、仮想環境をn分木構

造によって表現する。

【0004】例えば、図42は、ある簡単な仮想環境の例のイメージ図である。この図の場合、空間中の部屋、机、ソファに注目すると、世界座標系C0から、座標変換T2で変換された座標系C2上に部屋が記述され、その部屋の中の机やソファは、それぞれ座標変換T3、T4によって座標系C2から変換される座標系C3、C4上で記述されている。そして、机の上の壺は、座標系C3から座標変換T5によって変換される座標系C5上で記述されている。さらに、机の上に、光線空間データを配置している。これは、壺と同様に座標系C3から座標変換T6によって変換される座標系C6上で記述されている。これを、模式的な木構造で表現すると図43に示す木となる。

【0005】(2)3次元物体や空間の実写の画像群を基にして光線空間データを生成し、その光線空間データから、任意の位置・向きから見える画像を生成し、提示することによって3次元物体や空間を再現する方法があった。

【0006】また、複数の利用者間で協調作業を行うための方法として、

(3)複数の利用者が共有しているデータを記憶しておくデータ記憶装置と、データ記憶装置に対する処理を管理するためのサーバ装置102とを設け、各利用者がサーバ装置102と通信を行って、データ記憶装置内のデータを参照したり変更したりする、もしくは、各利用者間でデータ記憶装置内のデータの変更に關する通信を直接行うことで、お互いの持つデータを同一の状態に保つ方法があった。

【0007】さらに、協調作業を行う方法(3)と、仮想環境を表現する方法(1)とを利用して、

(4)複数の利用者間で、幾何学的形状データからなる仮想環境の状態を同一に保つことによって、仮想環境を共有する方法があった。

#### 【0008】

【発明が解決しようとする課題】しかしながら、上述した従来技術(1)～(4)では、以下のような問題点があった。

【0009】まず、従来技術(1)の方法は、形状が非常に複雑な物体に対してその形状データを生成する、あるいは再現することが困難であるという問題があった。さらに、形状が複雑な物体の形状データを実物体から3次元計測装置を用いて獲得することも困難であるという問題があった。特に、既にある複雑な形状や、複雑な表面の模様・反射特性（吸収透過特性）をもつような実物体を再現することは、より困難である。さらに言えば一般に人工物は表現しやすいが、自然物の表現が難しい。しかし、部屋や街並みと言った人工的な単純な主に平面から構成される3次元空間を表現することが、比較的少ないデータ量で行いやすいという長所を持っている。また、木構造によって仮想環境を表現する方法も、優れた

方法であるといえる。

【0010】また、従来技術(2)では、ある量の撮影画像から光線空間データを生成してしまえば、後は任意視点（正確には拘束条件がある）位置からの観察画像を生成可能である。これは、対象の形状を復元せずに、対象そのものの撮影画像を基に記述する方法であるため、対象を写実性高く表現することのできる優れた記述方法である。しかし、データ量が多く必要である、対象が広い範囲である場合、撮影が困難である等の観点から、どちらかというと広い空間を記述することよりも3次元物体を記述することに向いている手法であって、部屋や街並みのような3次元空間を表現するには向いていないという問題があった。

【0011】また、従来技術(1)と従来技術(2)は、それぞれ異なった特徴を持っているが、これまでは、従来技術(4)において、従来技術(3)と組み合わせられて共有仮想空間を実現するために用いられてきた技術は、従来技術(1)のみであった。そのため、従来技術(4)によって実現される共有仮想環境は、従来技術(1)の特徴を持つことになってしまい、自然物などの、形状が非常に複雑な物体を含むことができない仮想環境になってしまっていた。

#### 【0012】

【課題を解決するための手段】上記問題を解決するために、本発明に係る画像処理システムは、サーバと、該サーバに接続された複数の端末とからなる画像処理システムであって、第1の端末は、3次元的な位置と3次元的な向きと表面属性とを含む記述情報に基づいて、3次元物体の形状や3次元空間の形状を記述する第1の記述手段と、任意の視点からの3次元物体と3次元空間の見え方に従って、実写画像に基づいて、該3次元物体と該3次元空間を記述する第2の記述手段と、前記第1の記述手段と前記第2の記述手段による記述内容を構成要素とする木構造のデータを記憶する第1の記憶手段と、前記第1の記憶手段に記憶された木構造のデータに基づいて、仮想3次元空間の第1の位置から第1の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示する第1の表示手段と、前記第1の記憶手段に記憶された木構造のデータを、前記サーバに送信する第1の送信手段とを有し、前記サーバは、前記第1の送信手段により送信されたデータを受信する第1の受信手段と、前記第1の受信手段により受信されたデータを記憶する第2の記憶手段と、前記記憶手段に記憶されたデータを、前記サーバに接続されている端末に送信する第2の送信手段とを有し、第2の端末は、前記第2の送信手段により送信されたデータを受信する第2の受信手段と、前記第2の受信手段により受信されたデータを記憶する第3の記憶手段と、前記第3の記憶手段に記憶された木構造のデータに基づいて、仮想3次元空間の第2の位置から第2の方向を見た場合の、前記第1の記述手段

と前記第2の記述手段による記述内容を表示する第2の表示手段とを設けた。

【0013】また好ましくは、前記第1の端末は、前記第1の記憶手段に記憶された木構造のデータの構成要素の少なくとも一部に処理を施す処理手段と、前記処理手段による処理の内容に基づいて、前記第1の記憶手段に記憶された木構造のデータを変更する変更手段とを更に有し、前記第1の端末において、前記第1の送信手段は、前記変更手段により変更された木構造のデータを、前記サーバに送信し、前記第2の端末において、前記第2の表示手段は、前記変更手段により変更された木構造のデータに基づいて、仮想3次元空間の第2の位置から第2の方向を見た場合の、前記第1の記述手段と前記第2の記述手段による記述内容を表示する。

#### 【0014】

##### 【発明の実施の形態】

（実施形態1）図1に、実施形態1における共有仮想環境生成システムのブロック図を示す。

【0015】101は、共有仮想環境の状態を示すデータを保持するデータ記憶装置である。102は、共有仮想環境の状態を示すデータの変更を管理するサーバ装置102である。データ記憶装置101の内容は、サーバ装置102からの指示によってのみ、変更することができる。103から105は、本実施形態1において利用者に共有仮想環境の状態を表示し、共有仮想環境の状態を示すデータを変更する指示をサーバ装置102に対して送信するためのインターフェース装置を備えた端末装置である。

【0016】なお、本実施形態1においては、データ記憶装置101はサーバ装置102内にあるものとするが、サーバ装置102及び端末装置103～105が接続されているネットワーク上であれば、どこに存在していても構わない。

【0017】図2に、データ記憶装置101内のデータ構成を示す。

【0018】201は、初期ツリー状態を示す情報であり、共有仮想環境の初期状態を示すデータが、サーバ装置102の起動時に格納される。202から204は、共有仮想環境内に存在する物体の情報を含むオブジェクトレコードである。オブジェクトレコード202は、オブジェクトの番号を示すオブジェクト番号202a、オブジェクトをロックしているユーザーの番号を示すロックユーザー番号202bと、オブジェクトの世界座標系での位置や方向、または共有仮想環境を構成するツリーの中での位置や方向を示す位置データ202c1、オブジェクトの幾何学的な形状を示す形状データ202c2、オブジェクトの表面の反射係数などを示す表面属性データ202c3、オブジェクトの光線空間データ202c4などのデータを含む属性フィールド202cから構成される。オブジェクトレコード203、204も同じくオブジェクト番号、ロックユーザー番号、属性フィールドから構成される。

【0019】205は、最大オブジェクトレコード値であ

り、データ記憶装置101内に存在するオブジェクトレコードのうち、最大のオブジェクト番号が記憶される。

【0020】206から208は、仮想環境を共有しているユーザーの情報を含むユーザーレコードである。ユーザーレコード206は、ユーザーの番号を示すユーザー番号206a、ユーザーが状態を保持する最新のオブジェクトを示す最新オブジェクト番号206b、ユーザーが状態を保持する最新のユーザーを示す最新ユーザー番号206c、ユーザーの視点の位置や方向などの状態を示す属性フィールド206dから構成される。オブジェクトレコード207、208も同じく、ユーザー番号、最新オブジェクト番号、最新ユーザー番号、属性フィールドから構成される。

【0021】209は、最大ユーザーレコード値であり、データ記憶装置101内に存在するユーザーレコードのうち、最大のユーザー番号が記憶される。

【0022】図3に、サーバ装置102の内部構成を示す。

【0023】303は、処理手順を実行するCPUであり、304は、処理手順を記録しておく処理プログラム記憶装置である。305は、端末装置から送信されてきたコマンドを一時記憶しておくための先入れ先出しのコマンドバッファである。306は、端末装置からのコマンドを受信したり、端末へのデータ送信を行うために用いられるインターフェース装置である。また、307は、上述の各部を接続するためのバスである。

【0024】図4に、コマンドバッファ305の内部構成を示す。

【0025】複数の通信端末から送信されてきたコマンドが、コマンドバッファに順番に蓄積される。CPU303は、コマンドバッファの先頭から1つずつコマンドを取り出して処理を行なう。401、402、403は端末装置から送信されてきたコマンドである。

【0026】コマンド401の構造について説明する。ユーザー番号401aは、コマンドを送信した端末装置が持つユーザー番号を示している。返信先アドレス401bは、サーバ装置102がコマンドに対して返信を行なう際の送り先となるアドレスを示している。コマンド番号401cは、コマンドの種類を示している。ユーザーレコード401dは、コマンド401によって参照または変更されるユーザーレコードを示している。また、オブジェクトレコード401eは、コマンド401によって参照または変更されるオブジェクトレコードを示している。全ての種類のコマンドで全てのデータが必要なわけではなく、必要のないデータは省略することができる。

【0027】図5に、端末装置103の内部構成を示す。なお、端末装置104～105の内部構成も、端末装置103と同一のものであるとする。

【0028】501は、本実施形態1の端末装置側での処理手順を実行するCPUである。502は、処理手順を記憶しておく処理プログラム記憶装置である。503は、共

有仮想環境の状態を示すデータを保持するデータ記憶装置であり、サーバ装置102側のデータ記憶装置101とデータを共有する。504は、処理過程、処理結果を示す画像を、表示装置に送るために蓄えておくフレームバッファである。505は、フレームバッファ504にある画像を表示するための表示装置としてのウィンドウシステムである。506は、文字や処理過程での指示を入力するキーボードである。507は、画面上での位置を指定するマウスである。508は、サーバ装置102へコマンドを送信したり、サーバ装置102からのデータを受信したりするために用いられるインターフェース装置である。また、509は、上述の各部を接続するためのバスである。

【0029】図6に、データ記憶装置503のデータ構成を示す。

【0030】601、および603から610のそれぞれの構成要素は、図2に示したデータ記憶装置101のデータ構成における、対応する各要素と全く同じ意味を持つ。各端末装置のデータ記憶装置503の内容は、サーバ装置102と端末装置との通信によって、サーバ装置102側のデータ記憶装置101とほぼ同一に保たれている。ユーザー番号602には、端末装置の初期化の際に、サーバ装置102から与えられる番号が格納される。

【0031】それでは、まず、本実施形態1におけるサーバ装置102側の処理について、図7～図19のフローチャートを用いながら、以下に説明することにする。この処理プログラムは、処理プログラム記憶装置304に記憶されており、CPU303によって実行される。

【0032】図7に、本実施形態1におけるサーバ装置102の処理のフローチャートを示す。

【0033】始めに、ステップS701で、データ記憶装置101の初期化を行う。具体的には、全てのオブジェクトレコードおよびユーザーレコードを消去した後、初期ツリー状態を示す情報201を読み込む。この初期ツリー状態を示す情報201には、表面属性データを備えた形状データ、または光線空間データが含まれている。この初期ツリー状態を示す情報201は、予め何らかの既存のモデル等を用いて作成しておくことが可能である。

【0034】ここでいう形状データには、三角形パッチなどのポリゴンデータや、NURBSなどの自由曲面データなどがある。表面属性データとは、その形状の材質、反射特性や模様などであり、模様はテクスチャデータとして入力する。テクスチャデータとは、ポリゴンデータや自由曲面データなどの表面の模様を表すデータで、模様の画像データとその画像が貼る対象である形状とどういう位置対応となっているかの対応関係を記述したデータである。

【0035】また、光線空間データとは、図34に示すような座標系に $z=0$ の平面を仮定し、この平面（これを基準面34と呼ぶことにする）を通過する光線の集合として3次元空間を表現したデータである。

【0036】この光線空間データを用いた手法では、3次元空間内のzにある視点位置Pで観察できる画像は、この基準面34を通過する光線の集合からPを通過する光線のみをサンプリングして得られる画像と等価になる（図35参照）。一般的には、各光線は基準面34を通過する位置(x、y)、各光線がx軸、y軸のそれぞれとなす角度を $\phi$ 、 $\phi$ 、光線が平面を通過した時間t、光線の色(r、g、b)で表されるが、実際には、計算量やデータ量が膨大になるという問題から、対象は静止物体でy軸方向の視差はないと仮定されることが多い。この仮定のもとで $u = \tan n\phi$ とにおいて各光線をx-u空間に射影し、この射影されたx-u空間で光線空間を扱う場合、ある点Pを通過する光線は図36に示すように直線状の軌跡をなす。この軌跡は以下の式で表される。

$$\text{【0037】 } x = X - Z \cdot u \quad (1)$$

$$u = \tan \phi \quad (2)$$

ここで(X、Z)は観察視点位置を表し、xは光線がx-u空間上のx軸と交差する位置を表す。また、 $\phi$ は光線がz軸となす角度を表す。

【0038】まず、図37に示すように多数の視点位置で得られた画像から直線状の軌跡を求め、この軌跡群によりx-u空間が密に埋められていると仮定する。この時、zにある視点位置Qの画像は、図37に示すようにx-u空間上でQを通過する軌跡を求め、その軌跡上にすでに記録されている光線の色を逆に求めることにより得ることができる。

【0039】さて、次に、ステップS701でサーバ装置102の初期化が終了した後の処理について説明する。サーバ装置102は、端末装置からのコマンド待ちのループに入り、ステップS703で、コマンドバッファ305に端末装置からのコマンドが入力されていないかを確認する。コマンドバッファ305が空の場合には、再び端末装置からのコマンド待ちのループに入る。また、コマンドバッファ305が空でない場合には、ステップS704に進み、コマンドバッファ305の先頭にあるコマンドに従い、データ記憶装置101に対する操作を行う。操作の結果、端末装置に対してデータを送信する必要がある場合には、インターフェース装置306を通じてデータを送信する。ステップS704の処理が終了した後は、再び、端末装置からのコマンド待ちのループに入る。

【0040】ステップS704においてコマンドバッファ305から読み込むコマンドには、ユーザーレコード追加、ユーザーレコード削除、ユーザーレコード属性問合せ、ユーザーレコード変更、最大ユーザーレコード値問合せ、オブジェクトレコード追加、オブジェクトレコード削除、オブジェクトレコード属性問合せ、オブジェクトレコード変更、オブジェクトレコードロック、オブジェクトレコードアンロック、最大オブジェクトレコード値問合せの12種類のコマンドがある。上記の12種類のコマンドを受信したときのサーバ装置102の処理のフロー

チャートを、それぞれ図8から図19に示す。

【0041】図8に、ユーザーレコード追加コマンド受信時の、サーバ装置102の処理のフローチャートを示す。ステップS801で、データ記憶装置101内の最大ユーザーレコード値209を、1だけ増加させる。次に、ステップS802で、新しいユーザーレコードをデータ記憶装置101に作成する。この際、新しいレコードのユーザー番号には、最大ユーザーレコード値209を格納する。最後に、ステップS803で、最大ユーザーレコード値209と、初期ツリー状態を示す情報201を返信する。

【0042】図9に、ユーザーレコード削除コマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからユーザーレコード削除コマンドを受信すると、ステップS901で、データ記憶装置101に含まれる全てのユーザーレコードのロックユーザー番号を調べ、ユーザーUによってロックされているユーザーレコードのロッカーユーザー番号を0にする。最後に、ステップS902で、データ記憶装置101から、ユーザーUに対応するユーザーレコードを削除する。

【0043】図10に、ユーザーレコード属性問合せコマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUから、ユーザーレコードRの属性問合せコマンドを受信すると、ステップS1001で、ユーザーレコードRがデータ記憶装置101に存在するかどうかを確認する。存在すれば、ステップS1002で、ユーザーレコードRを返信する。また、存在しなければ、ステップS1003で、「消去」を表すデータを返信する。ステップS1002またはステップS1003の処理が終了すると、ステップS1004で、ユーザーレコードUの最新ユーザー番号を調べ、Rより小さければRに置き換える。

【0044】図11に、ユーザーレコード変更コマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからユーザーレコード変更コマンドを受信すると、コマンド中のユーザーレコードの属性フィールドをPとして、ステップS1101で、データ記憶装置101のユーザーレコードUの属性フィールドをPに置き換える。

【0045】図12に、最大ユーザーレコード値問合せコマンド受信時の、サーバ装置102の処理のフローチャートを示す。ステップS1201で、データ記憶装置101の最大ユーザーレコード値209を返信する。

【0046】図13に、オブジェクトレコード追加コマンド受信時の、サーバ装置102の処理のフローチャートを示す。ステップS1301で、データ記憶装置101内の最大オブジェクトレコード値205を1だけ増加させる。次に、ステップS1302で、新しいオブジェクトレコードをデータ記憶装置101に作成する。その際、オブジェクト番号が最大オブジェクトレコード値205に、属性がコマンドに含まれるオブジェクトレコードの属性となるようにする。

【0047】図14に、オブジェクトレコード削除コマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからオブジェクトレコードRの削除コマンドを受信すると、ステップS1401で、データ記憶装置101のオブジェクトレコードRがユーザーUによってロックされているかどうか調べる。ロックされていないければ何もせず、ロックされていればステップS1402でRを削除する。

【0048】図15に、オブジェクトレコード属性問合せコマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからオブジェクトレコードRの属性問合せコマンドを受信すると、ステップS1501で、Rがデータ記憶装置101に存在するかどうかを確認する。存在すれば、ステップS1502で、Rを返信する。存在しなければ、ステップS1503で、「消去」を表すデータを返信する。ステップS1502またはステップS1503の処理が終了すると、ステップS1504で、ユーザーレコードUの最新オブジェクト番号を調べ、Rより小さければRに置き換える。

【0049】図16に、オブジェクトレコード変更コマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからオブジェクトレコード変更コマンドを受信すると、ステップS1601で、データ記憶装置101のオブジェクトレコードRがユーザーUによってロックされているかどうか調べる。ロックされていないければ何もせず、ロックされていれば、コマンド中のオブジェクトレコードの属性フィールドをPとして、ステップS1602で、データ記憶装置101のオブジェクトレコードRの属性フィールドをPに置き換える。

【0050】図17に、オブジェクトレコードロックコマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからオブジェクトレコードRのロックコマンドを受信すると、ステップS1701で、データ記憶装置101のオブジェクトレコードRが0であるか調べる。0でないならば何もせず、0ならばステップS1702でオブジェクトレコードRのロックユーザー番号をUに書き換える。

【0051】図18に、オブジェクトレコードアンロックコマンド受信時の、サーバ装置102の処理のフローチャートを示す。ユーザーUからオブジェクトレコードRのロックコマンドを受信すると、ステップS1801で、データ記憶装置101のオブジェクトレコードRがUであるか調べる。Uでないならば何もせず、Uならばロックユーザー番号を0に書き換える。

【0052】図19に、最大オブジェクトレコード値問合せコマンド受信時の、サーバ装置102の処理のフローチャートを示す。ステップS1901で、データ記憶装置101の最大オブジェクトレコード値205を返信する。

【0053】なお、サーバ装置102は、以上に述べた12種類のコマンドに対する処理をコマンドバッファ305

にコマンドが格納されている限り繰り返す。

【0054】それでは、次に、本実施形態1における端末装置103側の処理について、図20～図24、図28、及び図33のフローチャートを用いながら、以下に説明することにする。この処理プログラムは、処理プログラム記憶装置502に記憶されており、CPU501によって実行される。

【0055】図20に、本実施形態1における端末装置103の処理のフローチャートを示す。

【0056】始めに、ステップS2001で、データ記憶装置503の初期化を行う。次に、描画ループに入る。ループ内では、まずステップS2002で、データ記憶装置503のオブジェクトレコードの更新を行う。続いてステップS2003で、データ記憶装置503のユーザーレコードの更新を行う。更新が終わると、ステップS2004で、データ記憶装置503のデータを用いてフレームバッファ504に対して描画を行う。描画が終わると、ステップS2005でキーボード506やマウス507からのユーザーの指示を受け取る。ユーザーが終了の指示を行った場合には、ステップS2006で終了処理を行う。それ以外の場合には、ユーザーの指示に従い、サーバ装置102にデータ記憶装置101の内容を変更するためのコマンドを発行し、再び描画ループに入り、ステップS2002に戻る。

【0057】図21に、初期化のステップS2001における詳細な処理のフローチャートを示す。

【0058】ステップS2101では、全てのオブジェクトレコードおよびユーザーレコードを消去する。次に、ステップS2102で、サーバ装置102に対して、ユーザーレコード追加コマンドを、返信先アドレスを端末装置103として発行する。ステップS2103で、サーバ装置102から、初期ツリー状態を示す情報201、およびユーザー番号が返信されると、データ記憶装置503で、それぞれを、初期ツリー状態を示す情報601、およびユーザー番号602として記録する。

【0059】図22に、データ記憶装置503のオブジェクトレコードの更新を行うステップS2002における詳細な処理のフローチャートを示す。

【0060】まず、ステップS2201で、最大オブジェクトレコード値問合せコマンドを、返信先アドレスを端末装置103のアドレスとして発行する。次に、ステップS2202でサーバ装置102から最大オブジェクトレコード値205を受信して、ステップS2203で受信した値をデータ記憶装置503の最大オブジェクトレコード値606と比較する。二つの値が等しければステップS2205へ、等しくなければステップS2204へ進む。

【0061】ステップS2204では、データ記憶装置503の最大オブジェクトレコード値606とサーバ装置102から受信した最大オブジェクトレコード値205との差の個数だけ新しいオブジェクトレコードを作成し、データ記憶装置503の最大オブジェクトレコード値606をサーバ装置10

2から受信した最大オブジェクトレコード値205に更新した後、ステップS2205へ進む。この際に、新しく作成するオブジェクトレコードのオブジェクト番号は、データ記憶装置503の最大オブジェクトレコード値606に1を加えたものから、サーバ装置102から受信した最大オブジェクトレコード値205まで、1刻みにつけるようにする。

【0062】ステップS2205でループ変数rを1にセットし、オブジェクト更新ループに入る。ステップS2206では、データ記憶装置503にオブジェクト番号がrであるオブジェクトレコードが存在するかどうかを調べ、存在しなければステップS2212へ、存在すればステップS2207へ進む。ステップS2207では、オブジェクトレコードrの属性問合せコマンドをサーバ装置102に発行する。続いてステップS2208でサーバ装置102からの返信を受信し、ステップS2209で受信した値が「消去」を表すデータであるかどうかを確認する。「消去」であれば、ステップS2210でデータ記憶装置503のオブジェクトレコードrを消去した後、ステップS2212に進み、「消去」でなければ、ステップS2211でデータ記憶装置503のオブジェクトレコードrを更新した後、ステップS2212に進む。

【0063】ステップS2212では、rの値を最大オブジェクトレコード値606と比較し、rの値が最大オブジェクトレコード値606以上であれば、ループを抜け、データ記憶装置503のユーザーレコードの更新を行うステップS203の処理へと進む。rの値が最大オブジェクトレコード値606未満であれば、ステップS2213でrの値に1を加えた後、ステップS2206に戻る。

【0064】このようにして、ステップS2002におけるデータ記憶装置503のオブジェクトレコードの更新が終了したら、ステップS2003でデータ記憶装置503のユーザーレコードの更新を同様に行う。

【0065】図23に、データ記憶装置503のユーザーレコードの更新を行うステップS2003における詳細な処理のフローチャートを示す。

【0066】このステップS2003のフローチャートでは、処理対象がオブジェクトレコードではなく、ユーザーレコードであるという点を除き、ステップS2002と同じ処理を行なうので、詳しい説明は割愛する。

【0067】図24に、データ記憶装置503のデータをもとに描画を行うステップS2004における詳細な処理のフローチャートを示す。

【0068】始めに、ステップS2401で、データ記憶装置503からオブジェクトレコードを読み込む。次に、ステップS2402で、全てのオブジェクトレコードの読み込みが終了したか判定し、終了していなければステップS2401に戻り、次のオブジェクトレコードを読み込む。また、読み込みが終了していれば、ステップS2403で、オブジェクトレコード中の属性フィールドに含まれる位置データに従い、オブジェクトを初期ツリー上に配置す

る。

【0069】次に、ユーザーレコードについて上記のステップS2401からステップS2403と同様の処理を行う。まず、ステップS2404で、データ記憶装置503からユーザーレコードを読み込み、次に、ステップS2405で、全てのユーザーレコードの読み込みが終了したか判定し、終了していなければステップS2404に戻り、次のユーザーレコードを読み込む。ユーザーレコードの読み込みが完了すると、ステップS2406で、読み込んだユーザーレコードを、ステップS2403で生成されたツリー上にユーザーレコードの属性フィールドに従い配置する。これにより、仮想環境を共有している他のユーザーが、仮想環境内でどの位置から、どの方向を観察しているかを確認することができる。最後に、ステップS2407で、ステップS2406で生成されたツリーを探索し、描画処理を行う。

【0070】本実施形態1においては、共有仮想空間を構築する際に、従来技術(1)と同様に仮想環境を構成する要素（座標変換データ、形状データ、表面属性データ、照明など）を木構造により表現する。これは、3次元空間を構成する空間、地面、建物、部屋、家具、照明、置物などは、それぞれもともと階層的な入れ子関係にあると考えられるためである。例えば、机の上の置物は、机の配置とともに動くように、机の配置に依存する関係にあり、机の存在する座標系から相対的に配置したほうが都合がよい場合が多い。そのため、階層的に配置に依存関係のあるデータ構造をとる。その方法として、仮想環境をn分木構造によって表現する。

【0071】図25は、ある簡単な仮想環境の例のイメージ図である。この図の場合、空間中の部屋、机、ソファに注目すると、世界座標系C0から、座標変換T2で変換された座標系C2上に部屋が記述され、その部屋の中の机やソファは、それぞれ座標変換T3、T4によって座標系C2から変換される座標系C3、C4上で記述されている。そして、机の上の壺は、座標系C3から座標変換T5によって変換される座標系C5上で記述されている。ここで、本手法では従来技術(1)の手法とは異なり、幾何学的データだけではなく光線空間データも用いて木構造に含まれる物体を記述する事ができる。図25の場合、机の上に光線空間データを配置している。このデータは、座標系C3から座標変換T6によって変換される座標系C6上で記述されている。

【0072】複数の利用者で仮想環境を共有する従来技術(4)では、光線空間データといった形状データとは異なるデータを混在して記述された仮想環境を共有することはできなかったが、本手法では、光線空間データも形状データと同様に、同じ木構造の一要素として記述し、その木構造を複数の利用者間で同一の状態に保つことで、複雑な形状を持つ物体を光線空間データとして表現し、部屋や町並みといったデータを形状データとして表現した仮想環境を、複数の利用者間で共有することがで

きる。

【0073】なお、図25に示した仮想環境を、模式的な木構造で表現すると、図26のようになり、また、この木構造データを、記録用としてテキストデータにより表現すると、図27に示すようなデータ構造となる。この図27において、「Separator」や「Translator」などのキーワードは、区別さえできれば他のものを用いてもよい。また、「{」「」による区切りも他の記号や文字列で置き換えても構わない。

【0074】このように、光線空間データを木構造を構成する一要素として記述したような仮想環境において、光線空間データを複数の利用者間で共有する場合、属性フィールドとして、光線空間データと位置データを持ち、位置データとして光線空間データが配置されている木構造上の位置と座標変換T6を持つようなオブジェクトレコードを通信することで、複数の利用者間で、仮想環境内での光線空間データを共有することができる。

【0075】図28に、このような木構造で表現された仮想環境においてツリーを探索して描画処理を行う、ステップS2407における詳細な処理のフローチャートを示す。

【0076】観察者が仮想環境を観察するためには、表示画面にその視点から仮想環境を見たときの見え方を描画する必要がある。図26に示すようなn分木のデータの木を、左ノードを優先して深さ優先探索にて全ノード辿っていくことにより、この木を構成するすべてのノードのデータにアクセスする。このとき、ノード内に書かれている情報をもとに、ノードにアクセスした時点で順次表示画面に描画していくことが基本的な方法である。

【0077】まず、ステップS2801にて、観察する画面の設定の取得を行う。次にステップS2802にて観察視点の位置・向きを、ユーザーレコードより取得する。それに引き続いて、ステップS2803に進み、木構造の探索を開始する。これは、上述のように左優先で深さ優先探索の方法をとる（ステップS2804）。この探索が終了するまでノードの探索を続ける（ステップS2805）。

【0078】そして、探索しているノードの内容が座標変換であった場合（ステップS2806）には、ステップS2807のカレント座標変換の処理に進む。これは、もし、今までに座標変換であるノードがなかった場合には、それを木のそのノードの深さよりも深い部分木の基本の座標変換としてカレント座標変換としておき、そうでなければ、既存のカレント座標変換にその新しく現れたその座標変換をかけることによってカレント座標変換を更新し、木のその深さより深い部分木でのカレント座標変換とする。木は深い方向への探索時には、現状のカレント座標変換を渡すことにより、次の深さへカレント座標変換を与える。深さ方向への枝が無くなり一段浅くなる時には、木の深いときに更新されたであろうカレント座標変換は浅い方向へは渡さない。以前にその深さの木を探

索しているときにあったはずであるカレント座標変換を用いる。

【0079】ステップS2806が偽の場合ステップS2808に進む。このステップにて、ノードの内容が表面属性を表す場合には、ステップS2809のカレント表面属性の処理に進む。これは、このノード現れた時点で木のそのノードの深さよりも深い部分木の基本の表面属性としてカレント表面属性とする。木の深い方向への探索時には、現状のカレント表面属性を渡すことにより、次の深さへカレント表面属性を与える。深さ方向への枝が無くなり一段浅くなる時には、木の深いときに更新されたであろうカレント表面属性は浅い方向へは渡さない。以前にその深さの木を探索しているときにあったはずであるカレント表面属性を用いる。

【0080】ステップS2808が偽の場合ステップS2810に進む。このステップにて、ノードの内容が幾何形状モデルを表す場合には、ステップS2811の幾何形状モデル描画処理に進む。これは、カレント座標変換を用いてその形状モデルの位置・向き・スケールなどの座標変換をし、カレント表面属性を用いて従来技術(1)に示した方法により表示画像に2次元の画像として観察者に提示する画面に描画を行う。このときに、提示画面の各画素に対応した奥行き値保存マップを用意しておく。そして、描画処理の時に、形状モデルを描画するとき、視点位置から見た画面の各画素に相当する位置の形状モデルの表面の3次元位置の奥行き値を、奥行き値保存マップの同じ画素に書き込む。このとき、もしも、すでに奥行き値保存マップがすでにこの処理によって書き込まれていたならば、新たに描画するために得られた奥行き値がすでに保存されている奥行き値よりも大きいときには、提示描画へのこの画素の書き込み、および、奥行き値保存マップの書き換えは行わない。

【0081】ステップS2810が偽の場合ステップS2812に進む。このステップにて、ノードの内容が光線空間データを表す場合には、ステップS2813の光線空間データからの描画処理に進む。これは、まず、カレント座標変換を用いて光線空間データの位置・向き・スケールなどの座標変換をする。そして、観察視点位置が、光線空間データの基準面からみて、どの位置・向きにあるかを計算する。この計算結果などから、光線空間データからその観察条件で生成すべき画像を生成する描画則を決定し、提示画面に表示する画像を生成する。このとき、幾何形状モデルからの描画の時と同様に、表示画面で画素単位での奥行き判定を行う。視点位置と光線空間データの基準面位置のおかれている位置との距離を光線空間データの奥行き値であるとし、この値と奥行き値保存マップに格納されている奥行き値との比較を行う。奥行き値保存マップの値よりも小さい画素には、光線空間データから生成された画像を上書きし奥行き値保存マップを更新する。そうでない場合は、そのまま何もしない。

【0082】ステップS2812が偽の場合、あるいはステップS2807、ステップS2809、ステップS2811、ステップS2813の処理が終わった場合、再び木の探索処理であるステップS2804に戻る。ステップS2805にて、木の探索が終わったと判定されたならば、描画処理を終了する。

【0083】更新されたオブジェクトレコード、ユーザーレコードをもとにした描画が終了すると、次に、ステップS2005で、ユーザーからの指示に従い処理を行う。この際、ユーザーが指示できることは、オブジェクトの追加、オブジェクトの削除、オブジェクトのロック、オブジェクトのアンロック、オブジェクトの属性変更、ユーザーの属性変更、終了の7種類である。

【0084】図33に、ユーザーからの指示に従って所望のコマンドを発行する、ステップS2005における詳細な処理のフローチャートを示す。

【0085】ユーザーがオブジェクトの追加を指示すると、ステップS3301で、ユーザーが予め作成しておいた、表面属性データを備えた幾何形状データまたは光線空間データが含まれるオブジェクトの属性を、コマンド中のオブジェクトレコードに入れる。ステップS3302でオブジェクトレコード追加コマンドをサーバ装置102に対して発行する。コマンドの発行が終了すると、ステップS2002に戻り、描画ループを再開する。

【0086】ユーザーがオブジェクトRの削除を指示すると、端末装置はステップS3303で、サーバ装置102に対して、オブジェクトレコードR削除コマンドを発行し、ステップS2002に戻る。

【0087】同様にユーザーが、オブジェクトのロックを指示、またはオブジェクトのアンロックを指示すると、それぞれステップS3304、ステップS3305でオブジェクトレコードロックコマンド、オブジェクトレコードアンロックコマンドを発行し、ステップS2002に戻る。

【0088】また、ユーザーがオブジェクトの属性変更を指示すると、ユーザーの行った操作に対応する変更をオブジェクトレコードに対して行い（ステップS3306）、変更を行ったオブジェクトレコードを含むオブジェクトレコード変更コマンドを発行する（ステップS3307）。

【0089】また、ユーザーが、視点移動など、何らかのユーザー属性の変更を引き起こすような操作を行なうと、ステップS3308で操作に対応した変更をユーザーレコードに加え、その後ステップS3309でユーザーレコード変更コマンドを発行する。

【0090】最後に、ユーザーが終了を指示すると、ステップS3310で、端末装置はサーバ装置102に対してユーザーに対応するユーザーレコードの削除を行うコマンドを発行して、処理を終了する。

【0091】サーバ装置102に接続されているデータ記憶装置101のデータを変更する、これらのコマンドは、データ内容によってはサーバ装置102に受け入れられな

いことがある。例えば、ロックしていないオブジェクトレコードの削除を指示しても、サーバ装置102はオブジェクトレコードの削除を行わない。しかし、端末装置は、自分の発行したコマンドがサーバ装置102によって受け入れられたかどうかを確認する必要はなく、コマンドの発行が終わるとすぐに次の処理へと移ることができる。

【0092】ここで、図29および図30に、ユーザー1とユーザー2が、ある仮想環境を共有している時の描画面面例を示す。図29は、ユーザー1の視点位置・方向に対応する仮想環境を、ユーザー1の端末装置の表示画面に表示したものであり、図30は、ユーザー2の視点位置・方向に対応する仮想環境を、ユーザー2の端末装置の表示画面に表示したものである。

【0093】この図29および図30において、それぞれのユーザーは、同一の仮想環境を異なる位置・方向から観察している。これらの図で、兎の縫いぐるみは光線空間データにより記述されており、部屋は形状モデルにより記述されている。また、図29においては、兎のオブジェクトをユーザー1がロックしていることを示すために、兎の周囲に、目印となるような幾何学物体が表示されている。さらに、図30においては、兎のオブジェクトをユーザー1がロックしていることを示すために、兎の上下に、ユーザー1の固有色を有する幾何学物体が表示されている。

【0094】次に、ユーザー1が、仮想環境中でロックした兎のオブジェクトに対して、位置を変更するような指示を行なうと、ユーザー1の端末装置の表示画面における描画面面例は、図29から図31のように変化し、また、ユーザー1による兎のオブジェクトの移動に従って、ユーザー2の端末装置の表示画面における描画面面例は、図30から図32のように変化する。

【0095】（実施形態2）実施形態1においては、光線空間データから表示画像を生成していたが、本実施形態2では、光線空間データを用いるかわりに、以下の方法により多視点画像から表示に必要な画像を生成する。

【0096】図38は入力された多視点画像から画像を再構成する原理を示す図である。この図のように、本実施形態2では多視点画像の撮影条件として、撮影方向に対して垂直な一直線上にカメラを並べて撮影しておく必要がある。図中、3801は被写体、3802は入力画像の撮影位置を結んだ撮影視点並び直線、3803は仮想カメラの仮想CCD面を表す仮想CCD、3804は観察者の位置( $x'$ 、 $z'$ )に置かれた仮想カメラである。

【0097】図39のフローチャートを用いて、多視点画像から画像を生成する処理を説明する。ステップS3901において注目ラインjを画像Pの先頭ラインにセットし、注目画素をjライン目の左端の画素にセットする。次に、ステップS3902において、画像Pのjラインi番目画素に対応した撮影視点並び直線3802上の画像Qの位置を

計算する。これは次のように計算できる。

【0098】物体中の一点Aが、視点位置Pの仮想カメラの画素位置Piに写っているとす。また、位置AとPを結ぶ直線が撮影視点並び直線3802と交差する位置をQとする。このとき、図38から、画素位置Piに写っているものは、視点位置Qで撮影した画像の画素位置Qiに写っているものと等価であることが分かる。図38の幾何的な制約条件から、視点位置QのX座標は(1)式で表すことができる。但し、スキャンラインの中央の画素位置を0番目の画素とする。また、仮想カメラ3804の画素ピッチをd、焦点距離をf、原点から撮影視点並び直線3802までの距離をgとする。

【0099】

$$x = x' + i \cdot d \cdot (g - z') / f \quad (3)$$

【0100】同様に視点位置Pの仮想カメラの画素位置Pjに写っているものは、記憶装置に記憶された視点位置Rの画像の画素位置Rjに写っているものと等価である。この方法により、例えば画像Qのjラインi番目の画素値を画像Pのjラインi番目の画素にコピーして画像を再構成した場合、再構成された画像が歪む、すなわち、画像中の物体が縦に伸びたり、縮んだりすることがある。そこで、これを解消するためにステップS3903において画像Pのjライン目に対応する画像Qのライン番号を求める。画像Qのライン番号の求め方法に関して、図40を参照しながら説明する。

【0101】図40は上記で再構成された画像の歪みを補正する原理を示す図である。図中、4001は被写体、4002は再構成したい視点位置Pの画像、4003は撮影視点並び直線3802上の視点位置ステップSの画像である。

【0102】被写体4001中の一点Bについて考える。点BがY軸に近いが、再構成したい視点位置Pの画像4002のz座標値Pz、撮影視点並び直線4002上の視点位置ステップSの画像4003のz座標値Szが十分に大きい、または、再構成したい視点位置Pの画像4002のz座標値Pz、撮影視点並び直線4002上の視点位置ステップSの画像4003のz座標値Szがほぼ同じ値であると仮定する。このとき、点Bから発する光線は再構成したい視点位置Pの画像4002中のmライン目と撮影視点並び直線3802上の視点位置ステップSの画像4003中のnライン目に記録される。そこで、仮想CCD3803の画素ピッチをd、仮想カメラ3804の焦点距離をf、仮想CCD3803のライン数をNとすれば、

$$Pz \cdot \tan \alpha = Sz \cdot \tan \beta \quad (4)$$

$$\tan \alpha = d \cdot (N/2 - m) / f \quad (5)$$

$$\tan \beta = d \cdot (N/2 - n) / f \quad (6)$$

となる。(4)式、(5)式、(6)式より、

$$n = N/2 + (m - N/2) \cdot Sz / Pz \quad (7)$$

が得られる。従って、再構成したい視点位置Pの画像4002のm番目のスキャンラインの値は、撮影視点並び直線3802上の視点位置ステップSの画像4003の(7)式で与えられるn番目のスキャンラインの値と等価になる。

【0103】そこで、ステップS3903での処理の後に、ステップS3904に進み、画像Pのjラインi番目の画素に、画像Qのnラインi番目の画素の値をコピーする。この処理により、撮影視点並び直線3802上でない視点位置の画像を再構成するときに画像が歪むという現象をある程度抑えることができる。

【0104】その後、ステップS3905に移り、注目ラインj中のすべての画素の処理が終了しているかどうかを判断し、終了していればステップS3907に移り、そうでなければステップS3906に移る。ステップS3906では、注目画素iを右隣の画素に移し、ステップS3902に戻る。

【0105】また、ステップS3907では、画像P中のすべてのラインに対して処理が終了しているならばこのサブルーチンを終了する。そうでなければ、ステップS3908において、注目ラインjを次のラインに移し、注目画素iをラインjの左端にセットした後に、ステップS3902に戻る。

【0106】このように、撮影視点並び直線3802上で微小間隔で撮影された多視点画像群が得られているとすると、上記の考え方にに基づき、すべてのスキャンラインについて同様のことを繰り返せば、撮影視点並び直線3802上でない視点位置の画像を再構成することができる。

【0107】(実施形態3) 実施形態1においては、図1に示すように、仮想環境の状態を示すデータの変更を、全てサーバ装置102を通して行うことで、各端末装置間で仮想環境を共有していたが、図41に示すような、サーバ装置102を用いない通信形態でも、同様のことが可能である。

【0108】図41では、端末装置4101から4104がネットワークを通じてバス4105で接続されている。それぞれの端末装置は、実施形態1における端末装置と同様の処理を行う。ただし、実施形態1でサーバ装置102に対して発行していたコマンドは、ネットワーク上の全端末装置に伝わるように設定する必要がある。さらに、それに加えて、他の端末装置から発行された全てのコマンドを受信し、端末装置自体が実施形態1でのサーバ装置102と同じ処理を、自端末装置のデータ記憶装置503内のデータに対して行なうように設定する必要がある。

【0109】(その他の実施形態) また、本発明では、実施形態1において、サーバ装置102側のデータ記憶装置101と、複数の端末装置に含まれるデータ記憶装置503とを、何らかの手段を用いて物理的に共有することによって、実施形態1のような、端末装置とサーバ装置102との間の通信を行うことなく、仮想空間の共有を行うことも可能である。

【0110】また、同様に、実施形態3において、複数の端末装置に含まれるデータ記憶装置503を、何らかの手段を用いて物理的に共有することによって、実施形態3のような、端末装置間の通信を行うことなく、仮想空間の共有を行うことも可能である。

【0111】

【発明の効果】以上詳述したように、本発明によれば、従来技術(1)～(3)とをうまく組み合わせることにより、単純な形状データだけではなく、形状が非常に複雑な物体を含む仮想環境を、複数の利用者で共有することができるといった効果がある。

【0112】また、このような共有仮想環境内で、ある利用者がある物体に対して何らかの処理を施すと、他の利用者が、その処理による仮想環境内の変化を容易に視認することができるといった効果がある。

【図面の簡単な説明】

【図1】実施形態1のブロック図である。

【図2】サーバ装置102に含まれるデータ記憶装置101の構成を示した図である。

【図3】サーバ装置102の構成を示した図である。

【図4】コマンドバッファ305の構成を示した図である。

【図5】端末装置の構成を示した図である。

【図6】端末装置に含まれるデータ記憶装置503の構成を示した図である。

【図7】実施形態1における、サーバ装置102の処理を示したフローチャートである。

【図8】ユーザーレコード追加コマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図9】ユーザーレコード削除コマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図10】ユーザーレコード属性問合せコマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図11】ユーザーレコード変更コマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図12】最大ユーザーレコード値問合せコマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図13】オブジェクトレコード追加コマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図14】オブジェクトレコード削除コマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図15】オブジェクトレコード属性問合せコマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図16】オブジェクトレコード変更コマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図17】オブジェクトレコードロックコマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図18】オブジェクトレコードアンロックコマンド受信時のサーバ装置102の処理を示したフローチャートである。

【図19】最大オブジェクトレコード値問合せコマンド

受信時のサーバ装置102の処理を示したフローチャートである。

【図20】実施形態1における、端末装置の処理を示したフローチャートである。

【図21】端末装置の初期化処理を示したフローチャートである。

【図22】端末装置のオブジェクトレコード更新処理を示したフローチャートである。

【図23】端末装置のユーザーレコード更新処理を示したフローチャートである。

【図24】端末装置の描画処理を示したフローチャートである。

【図25】実施形態1で生成する仮想環境の例のイメージ図である。

【図26】図25の仮想環境を木構造で表現した図である。

【図27】図26の木構造をテキストデータで表現したデータを示した図である。

【図28】木構造を観察者に提示する処理を示したフローチャートである。

【図29】実施形態1で、観察者1が観察中の仮想環境の描画例を示した図である。

【図30】実施形態1で、観察者2が観察中の仮想環境の描画例を示した図である。

【図31】実施形態1で、観察者1が観察中の仮想環境の描画例を示した図である。

【図32】実施形態1で、観察者2が観察中の仮想環境の描画例を示した図である。

【図33】端末装置のユーザー操作の処理のフローチャートである。

【図34】光線空間の原理を説明するための3次元空間の模式図である。

【図35】実空間中のある点とその点を通る光線の関係図である。

【図36】実空間中のある1点を通る光線が、光線空間であるx-u空間へ写像されたときの写像を示した図である。

【図37】実空間と光線空間の相互変換を表す原理図である。

【図38】実施形態2で用いる多視点画像からの画像生成原理図である。

【図39】実施形態2で用いる多視点画像からの画像生成のフローチャートである。

【図40】実施形態2の画像生成における縦方向の歪み補正の原理図である。

【図41】実施形態3のブロック図である。

【図42】従来技術(1)の仮想環境の例のイメージ図である。

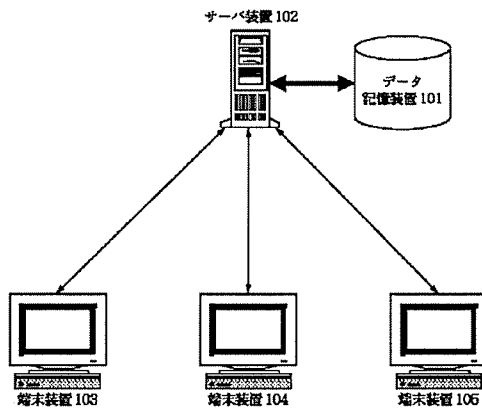
【図43】図42の仮想環境を木構造で表現した図である。

【符号の説明】

101 データ記憶装置  
102 サーバ装置102  
103 端末装置  
104 端末装置  
105 端末装置  
303 CPU  
304 処理プログラム記憶装置  
305 コマンドバッファ  
306 インターフェース装置  
501 CPU

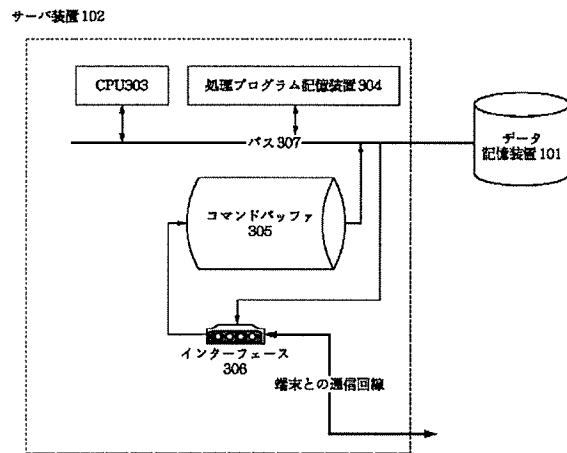
502 処理プログラム記憶装置  
503 データ記憶装置  
504 フレームバッファ  
505 ウィンドウシステム  
506 キーボード  
507 マウス  
508 インターフェース装置  
4101 端末装置  
4102 端末装置  
4103 端末装置  
4104 端末装置

【図1】



【図4】

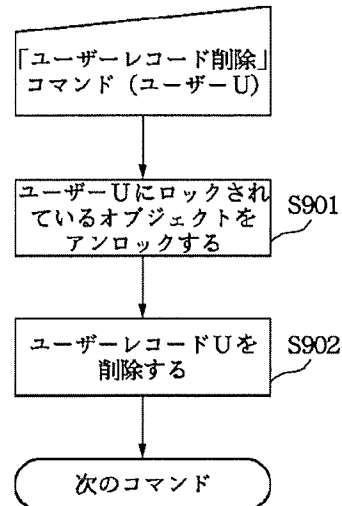
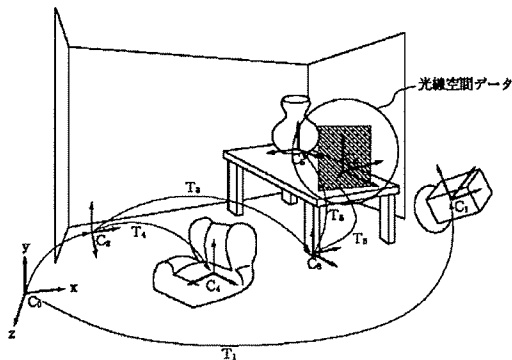
【図3】



【図9】

コマンド 401	ユーザー番号 401a	返信先アドレス 401b	コマンド番号 401c	ユーザー レコード 401d	オブジェクト レコード 401e
コマンド 402	ユーザー番号	返信先アドレス	コマンド番号	ユーザー レコード	オブジェクト レコード
コマンド 403	ユーザー番号	返信先アドレス	コマンド番号	ユーザー レコード	オブジェクト レコード

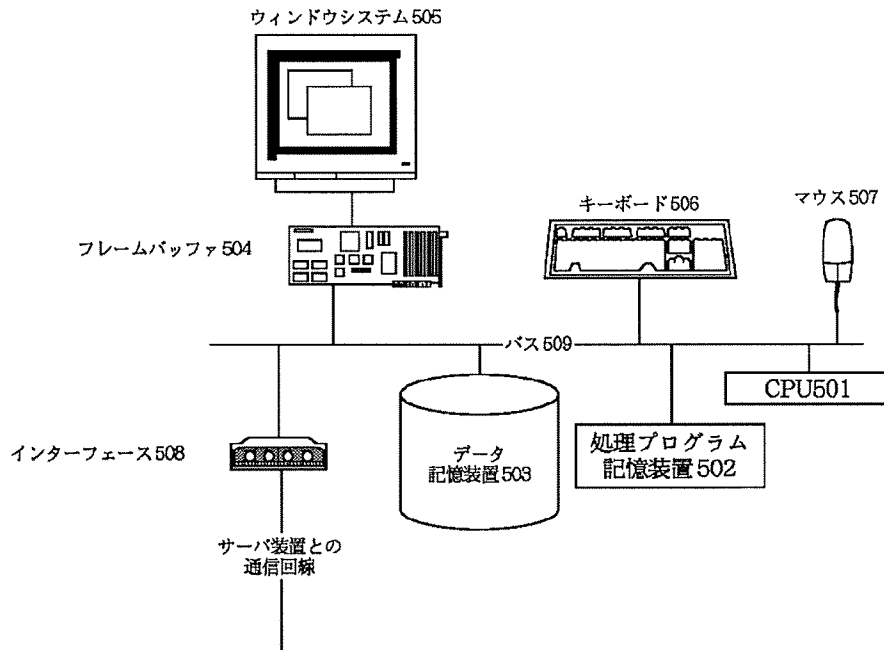
【図25】



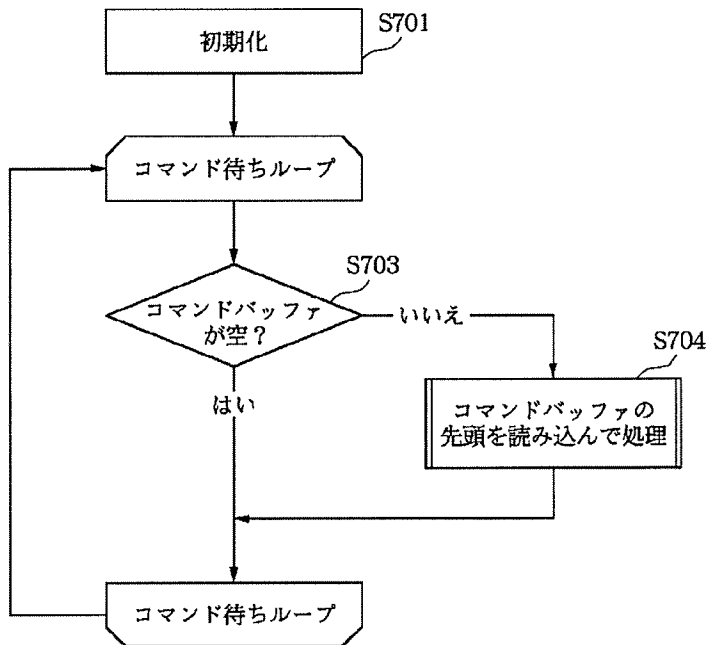
【図 2】

	初期ツリー状態201					
オブジェクトレコード202	オブジェクト番号 202a	ロックユーザー番号 202b	属性フィールド202c			
			位置データ 202c1	形状データ 202c2	表面属性データ 202c3	光線空間データ 202c4
						.....
オブジェクトレコード203	オブジェクト番号	ロックユーザー番号	属性フィールド			
			位置データ	形状データ	表面属性データ	光線空間データ
						.....
オブジェクトレコード204	オブジェクト番号	ロックユーザー番号	属性フィールド			
			位置データ	形状データ	表面属性データ	光線空間データ
						.....
	最大オブジェクトレコード値205					
ユーザーレコード206	ユーザー番号 206a	最新 オブジェクト番号 206b	最新ユーザー番号 206c	属性データ206d		
ユーザーレコード207	ユーザー番号	最新 オブジェクト番号	最新ユーザー番号	属性データ		
ユーザーレコード208	.....					
	ユーザー番号	最新 オブジェクト番号	最新ユーザー番号	属性データ		
	最大ユーザーレコード値209					

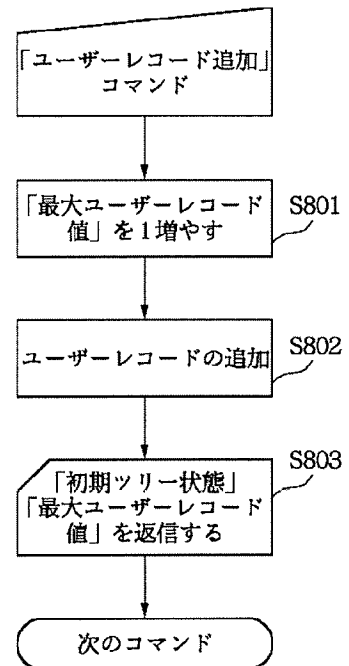
【図5】



【図7】



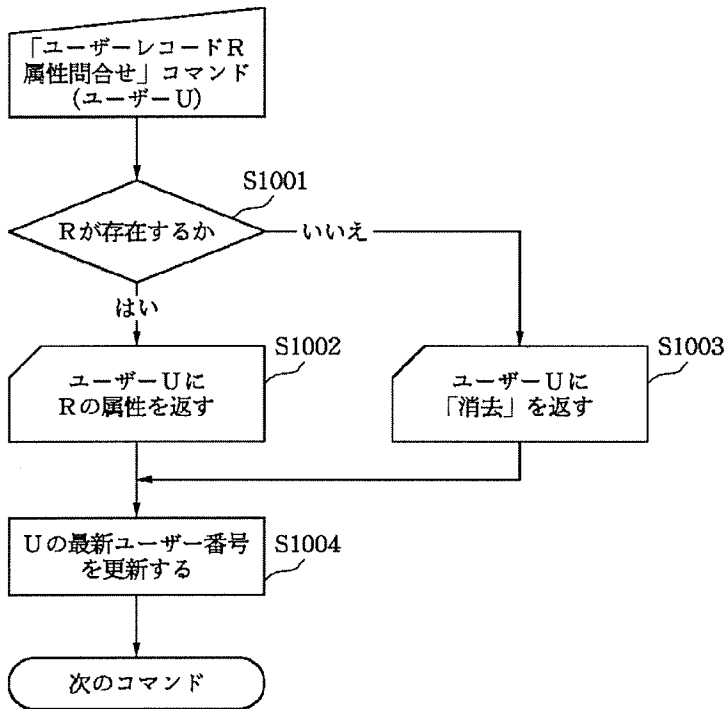
【図8】



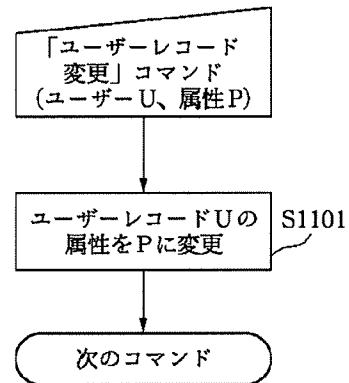
【図6】

	初期ツリー状態601				
	ユーザー番号602				
オブジェクトレコード603	オブジェクト番号 603a	ロックユーザー番号 603b	属性フィールド603c		
		位置データ 603c1	形状データ 603c2	表面属性データ 603c3	光線空間データ 603c4
					.....
オブジェクトレコード604	オブジェクト番号	ロックユーザー番号	属性フィールド		
		位置データ	形状データ	表面属性データ	光線空間データ
					.....
	.....				
オブジェクトレコード605	オブジェクト番号	ロックユーザー番号	属性フィールド		
		位置データ	形状データ	表面属性データ	光線空間データ
					.....
	最大オブジェクトレコード値606				
ユーザーレコード607	ユーザー番号 607a	最新 オブジェクト番号 607b	最新ユーザー番号 607c	属性データ607d	
ユーザーレコード608	ユーザー番号	最新 オブジェクト番号	最新ユーザー番号	属性データ	
	.....				
ユーザーレコード609	ユーザー番号	最新 オブジェクト番号	最新ユーザー番号	属性データ	
	最大ユーザーレコード値610				

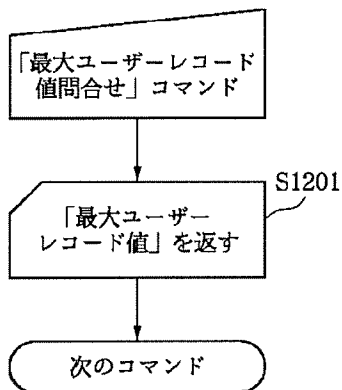
【図10】



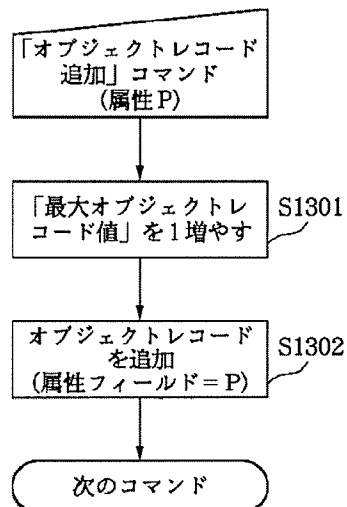
【図11】



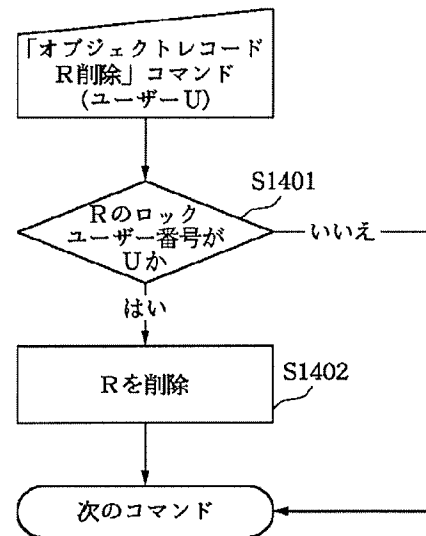
【図12】



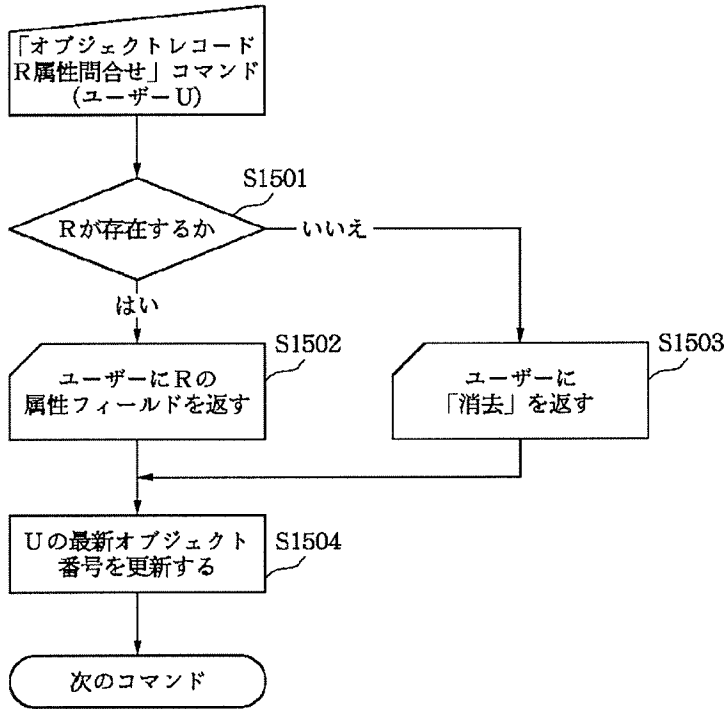
【図13】



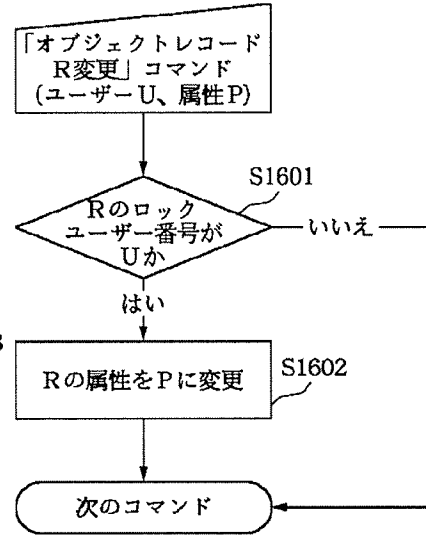
【図14】



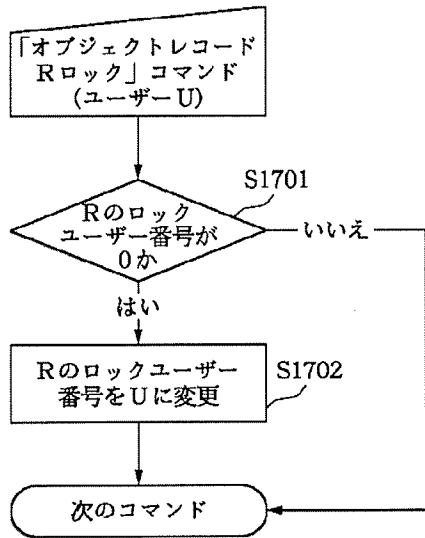
【図15】



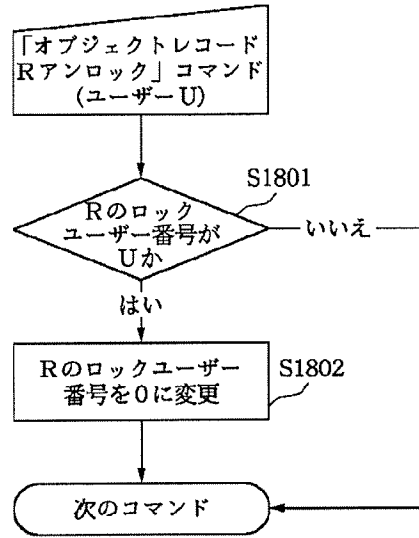
【図16】



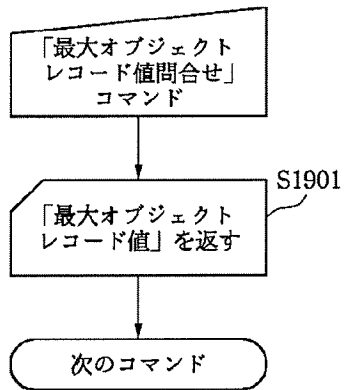
【図17】



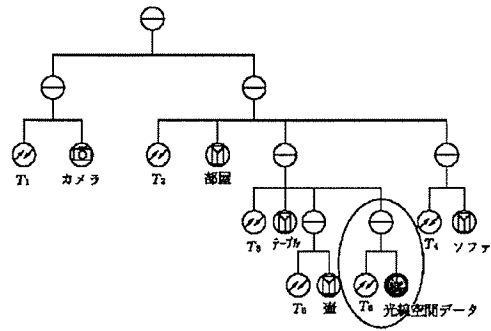
【図18】



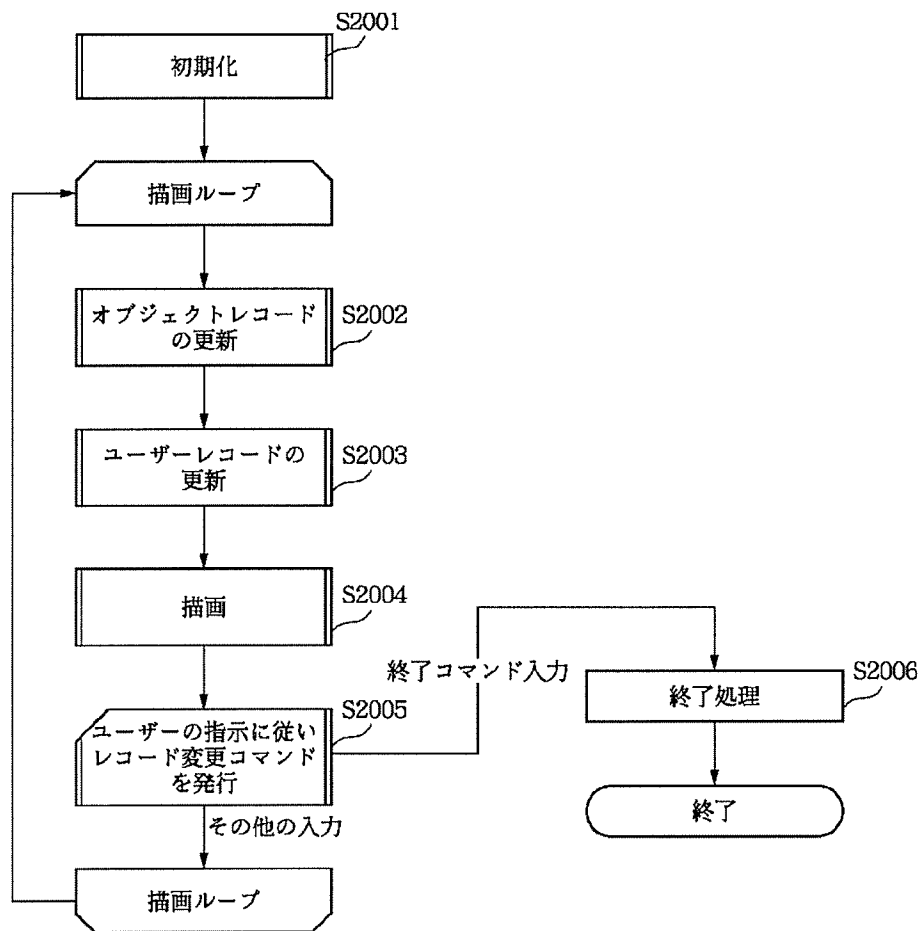
【図19】



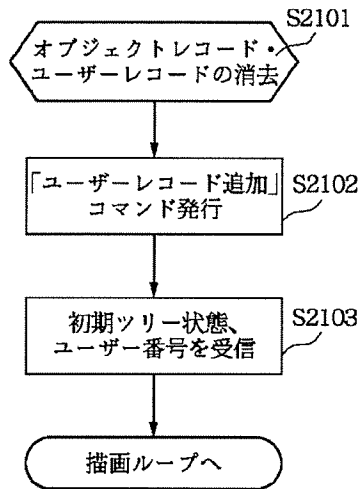
【図26】



【図20】

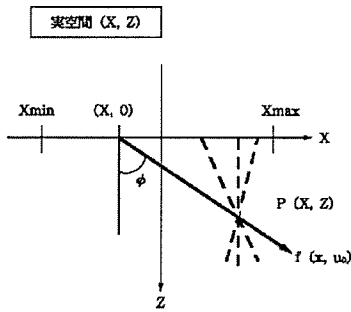


【図 2 1】

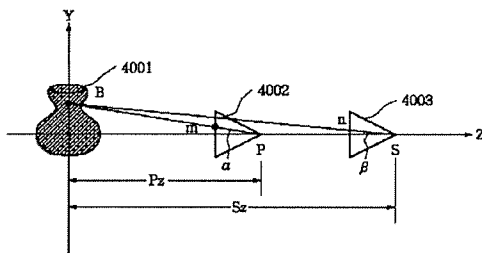


【図 3 5】

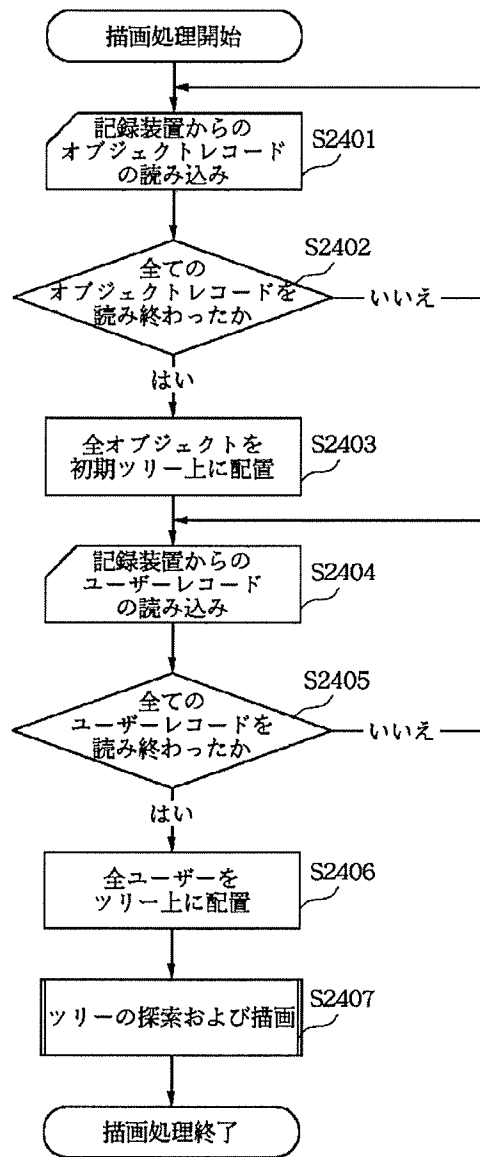
X 軸方向の 1 スキャンラインのみの場合



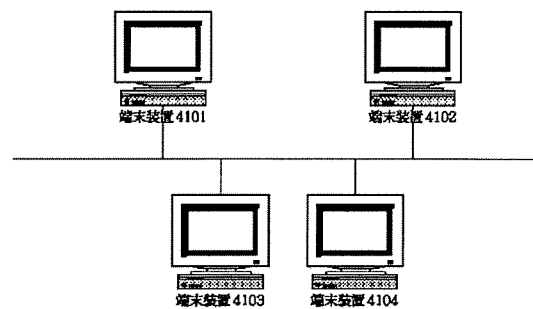
【図 4 0】



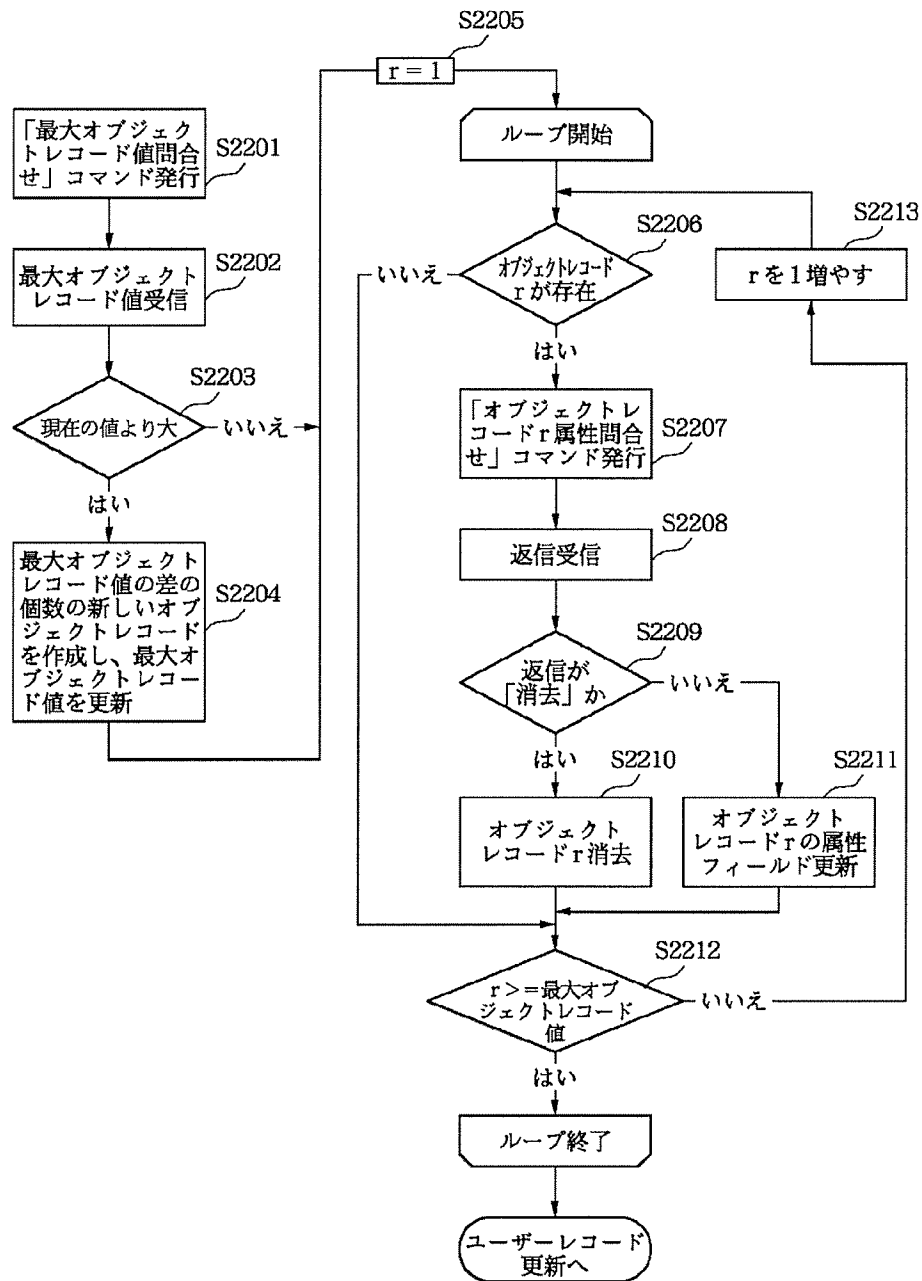
【図 2 4】



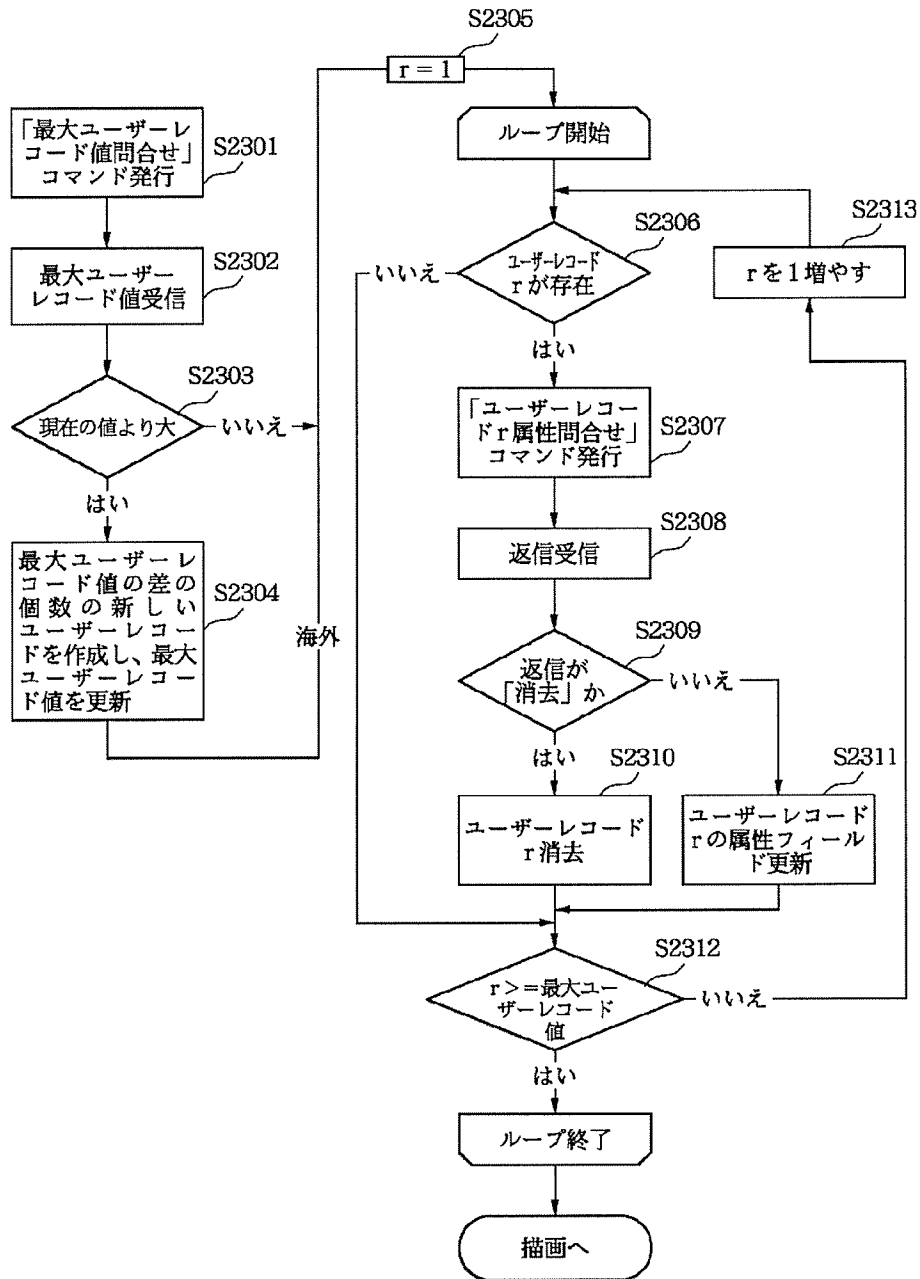
【図 4 1】



【図 2 2】



【図23】



【図 2 7】

```

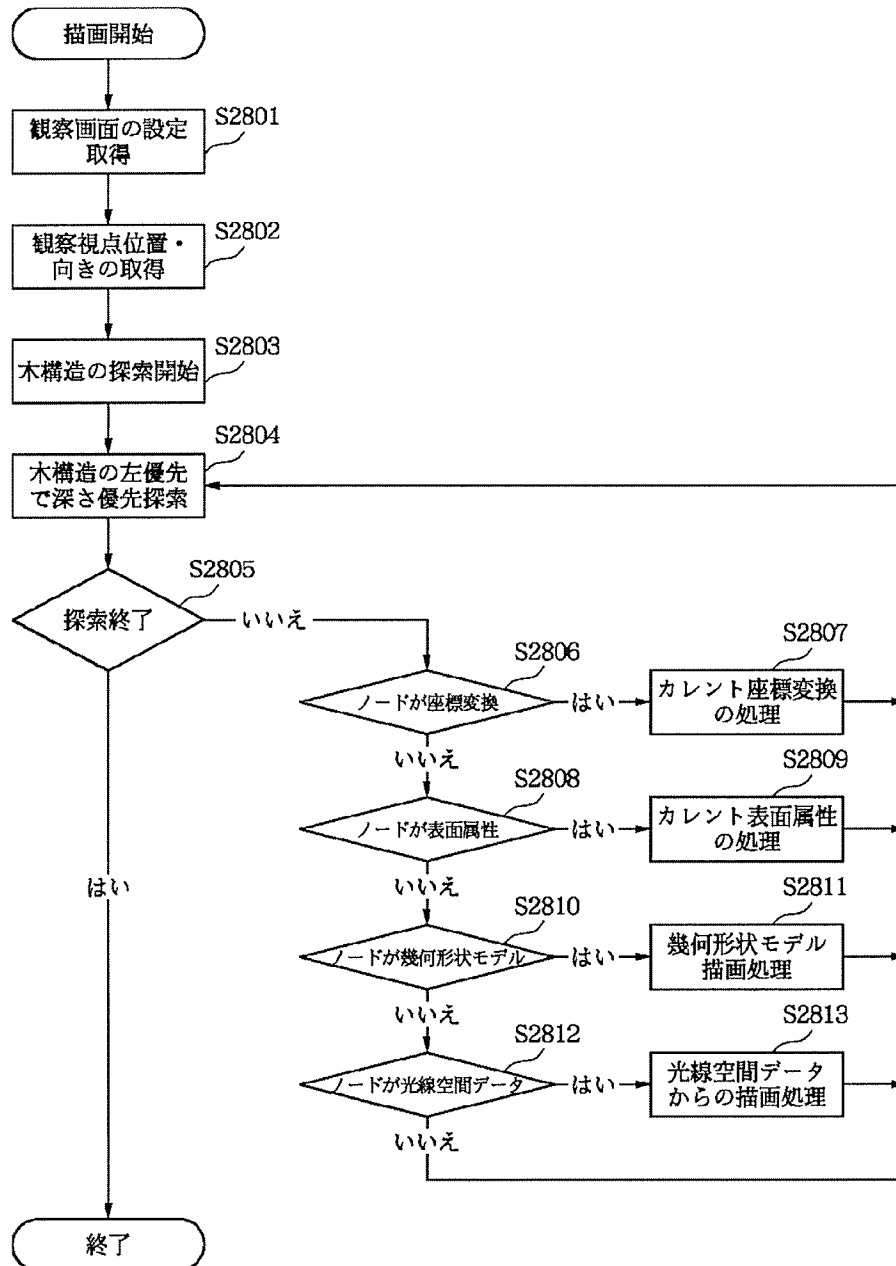
Separator { ← ルート
  Separator {
    Transform {
      座標変換 T1
    }
    Camera {
      位置, 画角などの情報
    }
  }
  Separator {
    Transform {
      座標変換 T2
    }
    ShapeModel
      部屋の形状
    }
    Separator {
      Transform {
        座標変換 T3
      }
      ShapeModel
        テーブルの形状
      }
      Separator {
        Transform {
          座標変換 T5
        }
        ShapeModel
          壺の形状
        }
      }
      Separator {
        Transform {
          座標変換 T6
        }
        光線空間データ {
          物体
        }
      }
    }
    Separator {
      Transform {
        座標変換 T4
      }
      ShapeModel
        ソファの形状
      }
    }
  }
}

```

Separator は部分木の先頭を表す

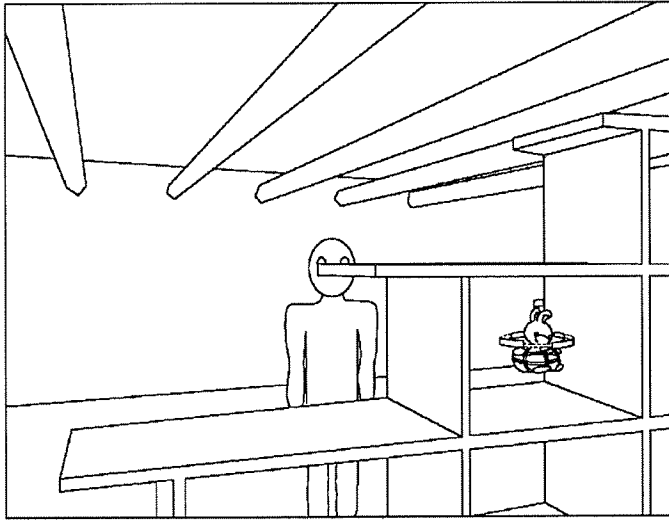
Separator や Transform などのキーワードは、  
 区別さえできれば他のどのようなものでもよい。

【図28】

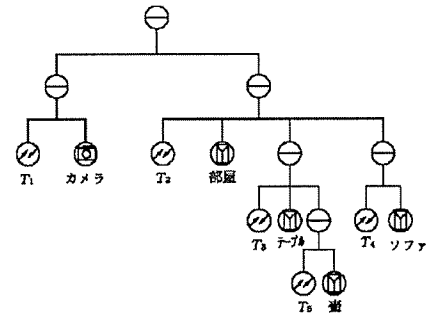




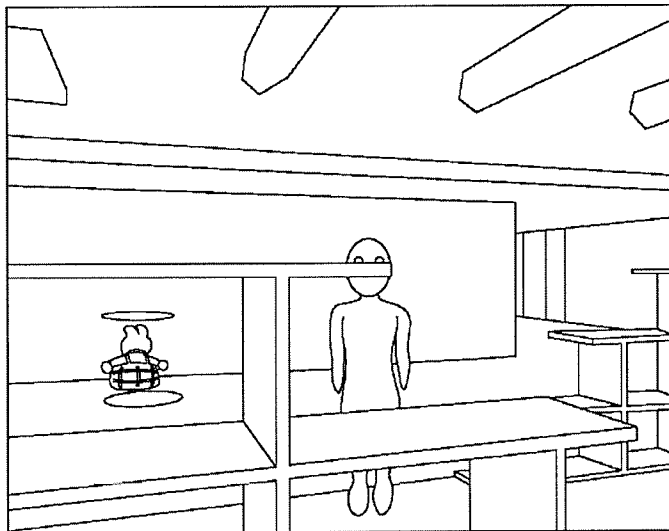
【図 3 1】



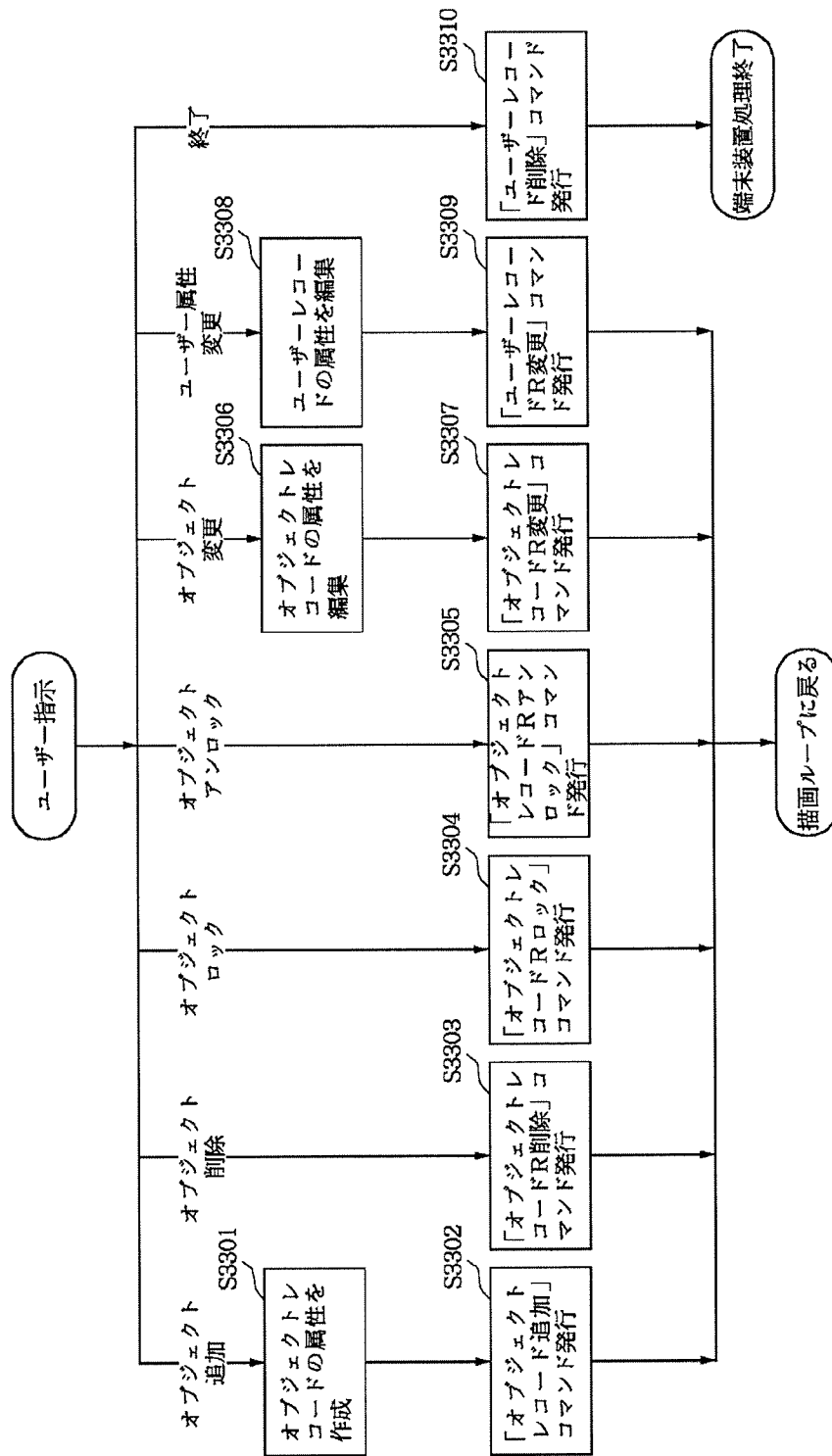
【図 4 3】



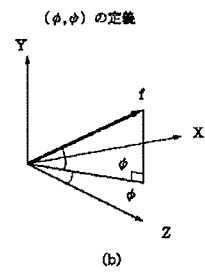
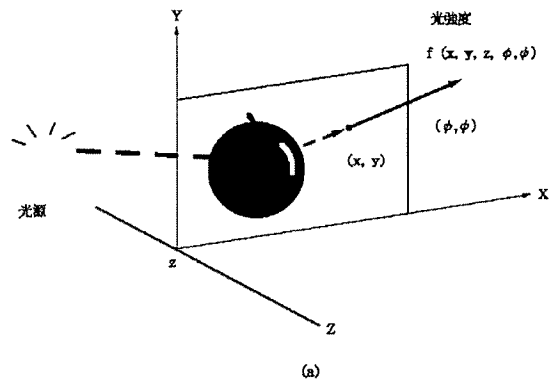
【図 3 2】



【図 33】

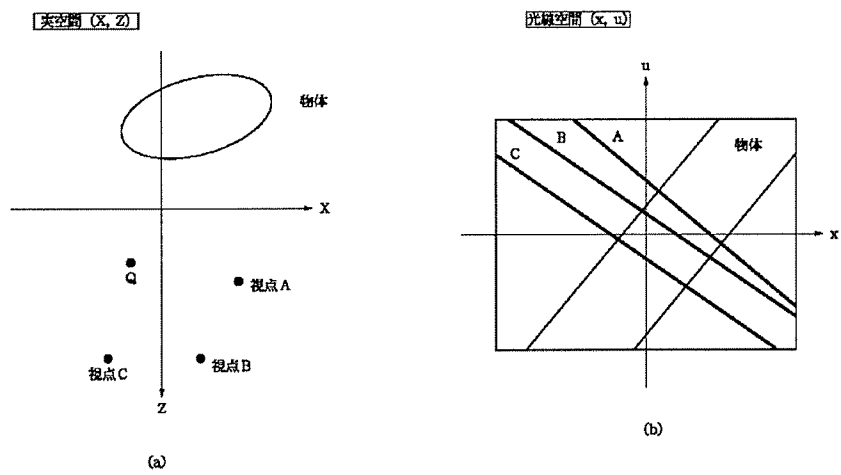


【図 3 4】



【図 3 7】

X 軸方向の 1 スキャンラインのみの場合



【図39】

